



Tech Info Library

Applesoft: PEEKs, POKEs, and CALLs

Revised: 1/20/87
Security: Everyone

Applesoft: PEEKs, POKEs, and CALLs

=====

To make Applesoft programs read data from memory, write data to memory, or pass control to machine language programs, programmers use Applesoft's PEEK, POKE, and CALL statements. Here is an explanation of each statement's function.

PEEK makes a program read a memory location. The format of the statement is PEEK (<memory location>) where <memory location> is a positive integer from 0 to 65535. Programmers use PEEK most commonly with a variable: X% = PEEK (2048) assigns the value located at 2048 to the integer variable X%.

POKE makes a program write a value to a memory location. The format of the statement is POKE <memory location>,<value> where <memory location> is a positive integer from 0 to 65535 and <value> is a positive integer from 0 to 255. Programmers use POKE most commonly to write data directly to memory: POKE 2048,128 assigns the value 128 to the memory location 2048.

CALL makes a program pass control to a machine language routine at some memory location. The format of the statement is CALL <memory location> where <memory location> is a positive or negative integer from -32768 to 32767 or a positive integer from 0 to 65535 (note that the signed integers from -32768 to 32767 represent exactly the same memory locations as the positive integers from 0 to 65535). Programmers use CALL most commonly used to invoke routines built into the Apple II's ROM. For example, the statement CALL -936 invokes the routine which clears the screen and homes the cursor (just like using Applesoft's HOME statement).

To change the screen display or make sounds and other special effects on the Apple II, Apple II Plus, Apple IIe, Apple IIc and Apple IIGS, Applesoft accesses various memory locations. Each particular CPU's reference manual includes a memory map where you can find the segments of memory used by text, graphics, Applesoft, the monitor and peripheral cards.

Apple-published memory locations remain the same for most members of the Apple II family; other internal locations may change. Therefore, to assure that your programs will work properly on all Apple II family computers, do not use entry points other than those printed in the Apple manuals.

Locations used to communicate with interface cards may be found in the manuals for those devices. For example, memory locations used by the Apple 80-column card are found in the 80-Column Text Card Manual and the Extended 80-Column Text Card Supplement.

Many computer and book stores sell books with listings of Applesoft, monitor ROM, DOS 3.3 and ProDOS memory locations. You may find the following publications useful:

- What's Where in the Apple by William Luebbert; Micro Ink.
- Beneath Apple DOS by Don Wirth and Pieter Lechner; Quality Software.
- Beneath Apple ProDOS by Don Wirth and Pieter Lechner; Quality Software.
- The Apple Almanac by Eric Goetz and Williams Sanders; Datamost, Inc.

Apple Technical Communications

Tech Info Library Article Number:56